Appendix for "Using Word Order in Political Text Classification with Long Short-term Memory Models" by Charles Chang and Michael Masterson

4.1 Weibo Data and Coding

We collect all geotagged Weibo posts from June 25, 2014 through June 15, 2015 posted by users in Beijing.[28] To distinguish political posts from non-political posts, we first develop a codebook to categorize a Weibo post as political, non-political, or unclear (Table 5). Posts are coded by trained coders who are native Chinese readers. The training and hand-coding takes place in three stages. We first take six random samples of 50 posts each and code these 300 posts according to our codebook. This enables us to introduce our coders to the codebook and coding process. We also divide our coders into pairs and ask them to code the same 100 posts. The intercoder reliability is high. Our coders label identical posts the same over 99% of the time (Krippendorf's $\alpha = 0.94$). Since this step is preliminary, we do not include any posts from this step in our final training sample.

The vast majority of posts are not political. In order to create a more balanced subsample to facilitate machine labeling, we create a list of 1350 phrases designed to capture every political post (see section 4.1.1). Because this list is designed to be over-inclusive, it also returns many non-political posts (60% of the posts filtered this way are not political). This use of keyword searches to select documents for further coding is similar to the special case of the "RIPPER method" described in D'Orazio et al. (2014). Using this method ensures that our hand coders will label enough posts of the less frequent category to train a classifier more quickly. It also allows us to introduce non-political posts we filtered out for the analysis with unbalanced data in section 3.4.

[28]The Sina Weibo Nearby Application Programing Interface (API) permits us to access the Sina Weibo database that stores all such posts. We collected more than 6 million unique posts from which we remove a small fraction that come from Apps that are unable to be verified, such as t.pp.cc.

Characters and phrases in our keyword list come from three sources. The first is the list of sensitive keywords on Weibo from Citizen Lab.[29] These keywords can be highly political, and users can be temporally banned from posting for using them. However, sometimes such posts are not deleted. The second source is the keywords used in King, Pan, and Roberts (2013). Third, we a use set of characters and phrases (some specifically related to anti-corruption events) brainstormed by our research team. They are complimentary to the first two sets of keywords in that they contain keywords that may indicate broad political issues and events.

We use this inclusive list of phrases to detect whether a post *may* be political. Even if a post contains only one phrase from our keyword list, it will be labeled potentially political. Otherwise, it is labeled non-political. Using another sample of 200 randomly drawn posts that do not contain any of the keywords on our list, we find that nearly all posts labeled as non-political are indeed non-political.[30] The potentially political posts, however, have a large share of posts that are non-political or unclear. We then ask four coders to code all posts that are labeled as potentially political into the three categories according to our code book and training. To protect our respondents and comply with IRB requirements, we drop 408 posts that contain personally identifying information from the dataset. The remaining posts coded in this step constitute our training sample in this paper. To be conservative in identifying political text, we combine the categories of non-political posts and unclear posts. After all posts are coded, we again ask our four readers to code the same 100 posts. The inter-coder reliability is still high, with each pair agreeing more than 94% of the time (Krippendorf's $\alpha = 0.88$).

[29]the list of sensitive keywords can be found at `https://chinadigitaltimes.net/china/sensitive-words-series/`.

[30]Over 95% of posts are non-political.

Table 5: Codebook for Weibo Posts

	Political	Non-political	Unclear or Neutral
Definition	Discussion of party and state leadership, for example, Xi Jinping; discussion of state institutions, such as the Constitution and legislature; discussion of state political campaigns, such as the anti-corruption campaign, ministerial reform, or propaganda; discussion of international relations that relate to China, for example, the "One China" policy; discussion of national economic, welfare, social, or religious policies; discussion of protest, petition, resistance against censorship, and other specific forms of political participation	Discussion of contents that are unrelated to politics, for example, personal relationships, personal emotions, entertainment, sports, games, selfies, travel, shopping, music and more. Discussion of political coursework, such as Marxist theory. Discussion of international relations that are unrelated to China, such as the Iraq issue or Iran deal. Jokes related to national leaders or their family members, but without political inclinations or implications	Discussion or statement without obvious political inclination or political attitude; Discussion of contents that can not be interpreted literally to reflect the individual's interest in politics; Discussion of social problems with indirect political meaning; Political discussion that are ambiguous or lacking necessary context

Table 5 – *Continued from previous page*

	Political	Non-political	Unclear or Neutral
Definition	讨论习近平等领导人，党，中央，国务院，以及国家政府机构，或国家政策（如宪法，反腐，国家机构改革，政治理论，政治口号等）。讨论国际关系，例如"一个中国"，明确表达支持或反对爱国主义。明确讨论关于经济，民生，社会，宗教，环境等政策法规。讨论具体的抗议示威，上访，讨论或挑战网络审查制度	讨论个人情感，娱乐圈，体育，游戏等与政治无关的内容，也包括自拍，旅游，购物，娱乐等。讨论政治课程，如马克思主义理论。讨论与中国无关的国际关系，如伊拉克问题。讨论与国家领导人或家人有关的笑话，但不带有政治倾向或暗示	没有明显的倾向，或者字面意思解读的政治态度明显不能反映个人的政治态度，与个人兴趣相关的国际政治，有间接的政治含义的社会问题，语意含糊，缺乏必要的语境，政治态度委婉无法过度解读

Examples of posts:

- **Political**: "常委没有免死牌，习总说到做到了，为习总店赞！" (The members of Standing Committee are not exempt from punishment. Well done, President Xi); "在一个不靠谱的体制内，面对这么一群不靠谱的人还要这么努力地靠谱才是最大的不靠谱！[思考][思考]" (In an unreliable political system, it is very much unreliable to work so hard when facing such a large group of unreliable people! [thinking hard] [thinking hard]); "转发泗水县苗馆镇马家井村老共产党员罗永吉，我的儿子罗才茂于1999年7月光荣的加入中国共产党，2000年开始任我们马家井村的党支部书

记，其死亡前刚刚参加了我们村新一届，但罗才茂的这些并不能改变其命运—2014年12月22日罗才茂同志光荣的去世在泗水县苗馆镇人民政府院内！” (Repost from Luo Yongji, an old Communist Party member of Majiajing Village, Miaoguan Town, Lishui County. My son Luo Caimao proudly joined the Communist Party of China in July 1999. In 2000, he became the party branch secretary of Majiajing Village. He just joined us [for the new party meeting] before his death. None of these can change Luo Caimao's fate—On December 22, 2014, Comrade Luo Caimao's 'gloriously' died within the People's Government Office of Miaoguan Town, Lishui County!)

- **Non-political:** ”#言短情长·晚安# 对弈的人已走,谁还在意推敲红尘之外的一盘残棋。” (“#Love in Haiku·Good night# The player who has played has gone, who is still interested in pushing a piece of chess outside the red dust.”); “青春总是在不经意间流走，而唯有时间会帮我们记忆那些最真最美好的人和事...” (Youth always flows inadvertently, and only time will help us remember the most beautiful people and memories...); “10号晴天，一大早就感觉又烤又蒸、大大的太阳晒着、只徒步了一站地实在是太热、时间也不早了改乘公交，下班太阳还很强烈、徒步回来竟然没出汗、也没感觉很热，晚饭后起风了刮的还不小、白天的炙热很快在风中消散、跳广场舞回来全身轻松、感觉比第一次跳的好。” (On the sunny day of the 10th, I felt that it was very hot in the early morning. It was too hot to walk, and it was too late to change to the bus. The sun was still very strong after work. However, it was not too bad to return home on foot. I did not sweat or feel very hot! After dinner, the wind blew, and the heat of the day quickly dissipated in the wind. I went for to Square dance, and I feel relaxed. I feel much better than the first time that I danced.)

- **Unclear or Neutral:** “#湖南军训教官师生互殴#希望挨打的学生都是有后台的！这些当兵的在部队肯定有人扛事，所以他们才不怕。一定要给学生一个说法！军训受累还得交军训费，现在居然还打学生！” (#Military trainer and trainee fight in Hunan# [I] hope that the students who are beaten have secret support! These soldiers

who are in the army must have some one at the upper level to support them, so they are not afraid to beat students. Be sure to give the students justice! Military training has to pay. To go as far as beating students!); "#抵制金秀贤#这TM是啥话题。。。。[挖鼻屎][挖鼻屎][挖鼻屎]【弱弱的说我觉得金秀贤还是挺萌的" ("#Protest Kim Soo Hyun# This is a topic of WTF...[[picking nose][digging booger][digging booger][I think Kim Soo Hyun is quite cute")

4.1.1 Keywords to Identify Potential Political Posts

国,带,周,法,性,微博,假,市,抓,区,院,微信,康,善,纪,恶,雾霾,影响,精神,省,自由,新闻,严重,美国,系统,领导,罪,网络,正能量,盼,教育,判,独立,污染,师傅,党,不明,购物,白云,评论,政府,新浪,组织,县,珍贵,消费,祖国,罚,雷锋,带走,公平,总算,稳定,主任,高级,涨价,公开,法律,释放,权利,老百姓,送礼,万元,待遇,爱国,革命,辞职,阅兵,有期,群众,清风,部队,屏蔽,改革,腐败,书记,新华,老朋友,财经,毛主席,洗脑,干部,正义,苍蝇,不当,权力,秘书,师父,户口,犯罪,官员,法院,江山,底层,社会主义,将军,总理,抗日,民主,抵制,贪官,广电,柴静,红旗,领导人,拆迁,有关部门,执法,富二代,死刑,民生,毛泽东,新闻联播,歧视,解放军,维权,李宁,总经理,防范,物价,民政,民众,屠杀,局长,非法,中级,纪委,阶级,贪污,市长,黑社会,郭美美,谷歌,审判,蛤蟆,特权,科学院,染指,统治,穹顶,政协,开除,普京,文革,美帝,翻墙,挺周,违规,马克思,钓鱼岛,康师傅,李佳,穹顶之下,食品安全,面瘫,恩恩,游行,涉嫌,影帝,犯法,纪律,政委,宪法,贫富,司令,秘书长,李明,审查,潜规则,强拆,村官,红军,市委,理想主义,选举,调动,受贿,讨薪,立案,常委,执政,金库,教育部,支书,验收,高高在上,纳税人,证监会,禁烟,公知,户籍,转世,领土,职务,戒严,砖家,陈鹏,新常态,快播,贪腐,落马,果敢,阻挠,邪教,审批,渣浪,巨额,熙来,封号,推特,贿赂,难民,迫害,水军,纪检,走私,获悉,检察院,环球时报,言论自由,打虎,跪舔,批斗,维稳,水表,参谋,杨光,社交网络,李超,王健,独裁,王爱民,芮成钢,徒刑,小金库,才厚,李小鹏,政务,老根,曼德拉,监察,省长,少将,羊群,巡视组,希特勒,形式主义,徐才厚,彭麻麻,政治家,销号,红歌,油管,张俊,金三胖,打黑,李刚,廉洁,廉政,政客,李浩,平反,网警,普世,沈佳,站中,安乐死,斯巴达,网信,王强,通奸,情妇,逮捕,河蟹,行贿,常委会,镇压,总参,清官,最高人民法院,美分,抄家,边飞,接受组织调查,张健,政治局,职权,路透,马英九,议会,禁言,老蒋,脸书,涉嫌严重违纪,贪污受贿,薛蛮子,贩

毒,弄虚作假,斯大林,自焚,林彪,观海,方舟子,宋祖英,开除党籍,屁民,最高法院,李鹏,小贪,无期徒刑,查水表,谢超,张宏,计生委,选民,张华,总后,蓝军,李军,枪支,敌对势力,敏感词,中级人民,希拉里,约谈,莫迪,涉嫌严重违纪违法,政法委,网评,收受,普世价值,无神论,监察部,赃物,民不聊生,谋取,封建社会,老习,普选,王岐山,官媒,李真,立案审查,领空,杨佳,滥用职权,范仲,质检总局,达赖,王斌,赵敏,杨生,民进党,蟾蜍,领导小组,猎狐,键盘侠,学运,彭德怀,战犯,王敏,集权,高鹏,裸官,最高人民检察院,收缴,接受调查,纽约时报,陈博,最高检,父母官,红二代,执行死刑,朱琳,网特,统战部,老朽,彭妈妈,提起公诉,杨波,柯文哲,歌功颂德,特首,贵国,李路,毒奶粉,臣子,赵俊,支那,张锐,极权,蒋经国,藏毒,李喜,面霸,立案侦查,艾未未,杨卫泽,违法违纪,黄菊,网宣,常文,开庭审理,膀胱癌,自干五,陈刚,立案调查,赃款,马克吐温,马壮,官民,收受贿赂,李达,杨华,红头文件,罗克,胡耀邦,蔡英文,强征,紧掏,陈岚,马超群,王昕,违宪,毛新宇,证物,路边社,非法经营,季建业,张鸿,拆哪,李庄,王安顺,谷俊山,郭金龙,非正常死亡,审计署,毛左,沙皇,涉嫌犯罪,财产公开,家奴,挺薄,,涉嫌受贿犯罪,王君,监察局,西方价值观,财厚,赵军,彭阿姨,戈尔巴乔夫,挪用公款,曾庆,谷开来,资中筠,非死不可,安理会,张兵,徇私枉法,死猪肉,王立军,米帝,索贿,铁流,常务副主席,思八达,曾平,李东生,梁振英,违纪问题,温相,统战部部长,邓大人,孙江,市检察院,张国强,沈冰,苏荣,黄丝带,朱明国,殖民主义,法广,涉嫌贪污,白恩培,罢免,西北狼,赵山,暴敛,曾飞,梁斌,粪青,胡德平,邹平,孙平,官方声明,左倾,张曙光,张高丽,涉嫌违规,贪污罪,定价权,张德江,数罪,权力斗争,杜伟,杨辉,河北省纪委,涉嫌受贿罪,王民,蒋洁敏,藏独,铁帽子,干涉内政,捌玖,收受巨额贿赂,教宗,李志鹏,杜怡,王永春,皿煮,秦玉海,编译局,学潮,开除党籍处分,张炬,方校长,李建民,林昭,涉嫌违纪,王海峰,申纪兰,禁评,组织部部长,美领馆,苏浩,贾庆林,赵紫阳,追缴,退党,高小燕,黄琳,山西省委常委,巨额财产来源不明,广东省政协主席,广州市委书记,彭嘛嘛,彭家声,提起公诉的,李春城,李珠,李长春,梁滨,汤灿,河北省委常委,湖北省纪委,王小玲,王洪钟,立案侦查并采取强制措施,罗欧,郭振玺,陈卫民,魏俊星,黄之锋,黄尧,宋林,山东省委常委,山西省纪委,张学军,张金泉,我裆,房国兴,方仁,木子月月鸟,李小琳,李志江,李英杰,杜善学,杨信,杨进先,栗战书,梁光,汪阳,河南省纪委,济南市委书记,浦志强,涉嫌贪污受贿,潘逸阳,王侃,王天朝,目田,立案检查,胡春华,自己选自己,蒋尊玉,薄瓜瓜,西朝鲜,财产公示,郑治,防火长城,陈安众,马向东,马昆,黄小虎,黄晓炎,黄汉,孙鸿志,宋建国,射击师,山西省人民检察院,广东省纪委,开除党籍

和公职,张中生,张凤国,张新华,张秀萍,报禁,撑起雨伞,撤销党内职务,时小雨,昆明市委书记,曾成杰,朱熔基,朱耘,李光熙,李兴华,李学友,李文昌,李柱,李量,李铁柱,杨晓波,武文元,毒大米,江省纪委,江绵恒,江苏省委常委,江苏省高级人民法院,江西省人大常委会,沈大伟,涉嫌贪污罪,涉嫌违纪违法,温宝,王丽娟,王宝军,王海鸣,王立新,玩忽职守罪,用职务上的便利为他人谋取利益,白培中,红朝,群蛆,聂春玉,肖念,舒刚,苏智,许杰,赵进喜,连子恒,郑怡,陈仲怀,陈港,隋凤富,黑老板,孙治,孟晓灵,孟钢,宋铜,宗新华,射击湿,总,山东省纪委,山东省纪委监察厅网站,山西省副省长,山西省纪委监察厅,巫向前,市三中院,广东省人民检察院,广东省委常委,广州市中级人民法院,广西壮族自治区纪律检查委员会,庆亲王,弄虚作假套取国家科技重大专项资金,张俊明,张哲英,张宝玉,张笑东,张雪忠,彭刚,彭辉,徇私枉法罪,总湿,成都军区联勤部部长,成都市中级人民法院,成都市纪委,才帝,打鸭,敢想不敢说,文家碧,新疆维吾尔自治区人民检察院,昆明市纪委,暂停履行职务,曹兴龙,朱吉祥,朱耀辉,朱锦华,李亚力,李全保,李崇禧,李德胜,李玉军,李长根,杨森林,欧阳坤松,欺骗组织,毛志刚,民煮,民猪,江苏省人民检察院,江苏省委原常委,江西省人大常委会副主任,江西省副省长,江西省纪委,沈培平,河北检察机关,浙江省军区副政委,涅姆佐夫,涉嫌严重违法,涉嫌职务犯罪,涉嫌贪腐,温党,温夫人,温帝,湖北省人民检察院,湖北省副省长,湖北省政协原副主席,湖北省纪委监察厅,湖南省人民检察院,湖南省纪委,王会师,王俊英,王兆军,王志忠,王振坤,王树新,王贵海,瓷器国,甘肃省人大常委会副主任,田勇,田玉林,申维辰,电婊,盐铁专卖,砖员,祝作利,福建省副省长,福建省纪委监察厅,秦建孝,秦皇岛市纪检机关,纽时,网络封锁,老贪,自治区政府副主席,蒋国星,薛万东,衡阳市纪委,被免去,西城区法院,西宁市委书记,解放军信息工程大学副政委,许栋,谢卓浩,谢鹏飞,谭栖伟,贵州省纪委,贾公子,赖昌星,赤匪,赵建华,辛希乐,违反中央八项规定精神问题,邓全忠,邓瑞祥,郭忠实,重庆市人大常委会副主任,陆四,陆肆,陈光C,陈弘平,陈柏槐,陈铁新,青海省委常委,韩先聪,驻马店市中级人民法院,高学文,高烈卿,鸟官,黄健骅,黄福明,电婊,孙屹峰,孙泽龙,孙造顺,孟照勤,安俊生,安小予,安德武,安荣辉,宋小林,宋维武,宜昌市监察局,宜昌市纪委,宣艮,宿州市桥区法院,屈礼仁,屎吧嗒,屎粑大,山东省东营市人民检察院,山西省临汾市纪委,山西省人大常委会副主任,山西省军区原司令员,山西省军区政治部原主任,山西省吕梁市 5 名干部,山西省太原市纪委,山西省政协副主席,山西省阳泉市检察院,岐山县纪委,工作日中午饮酒,帮助他人协调建设工程项目和工程款拨付,帮助申请长途客

运线路,常荣华,接受礼金,无偿占用,幸敬华,广东检察机关,广东省广州市人民检察院,广东省纪委常委,广东省纪委常委会,广州中院,广州军区空军后勤部部长,广州军区联勤部原副部长,广州市检察院,广西壮族自治区人民检察院,广西壮族自治区横县检察院,广西壮族自治区纪委,广西自治区政协副主席,庄孟虎,庆丰三年,庆丰二年,庆丰元年,庆丰帝,庞建春,庞铁华,康晓剑,廊坊检察院,廖少华,延庆法院,开除党籍开除公职处分,开除党籍处分将其涉嫌犯罪问题及线索移送司法机关依法处理,开除党籍处分按照干部管理权限,张世军,张东阳,张先玲,张其富,张友仁,张双娥,张学民,张彦春,张德利,张忠元,张文明,张文江,张文骏,张明全,张晓琪,张智江,张某周某夫妇,张淑芬,张燕南,张玉开,张秀利,张绍明,张肖平,张育浩,张豫生,张连德,张道亮,张金维,徐亚俊,徐卫东,徐双永,徐增增,徐孟加,徐尚红,徐建龙,徐忠岭,徐有,徐玉锁,徐立新,徐荣臻,忻州市中院,怀远县检察院,成都军区副司令员,成都市人民检察院,戚红伟,戴春宁,戴晓明,扭腰times,报销公款旅游费用,抹黑习,金道铭,拱产,挡中央,挪用公款收受贿赂,捌九,接受娱乐陪侍,接受宴请,接受补助,接受请托,提前透露拍卖信息违规发布公告指使他人假意参与竞买,撕八大,撕巴大,撤销中共福建省地震局机关党委专职副书记职务处分,撤销党内职务行政撤职处分,擅自脱岗,收受他人礼品,收受他人贿赂,收受医疗器械回扣,收受医疗器械回扣案件,收受巨额贿赂与他人通奸,收受巨额贿赂违反廉洁自律规定,收受巨额贿赂等严重违纪违法问题,收受或索取贿,收受或索取贿赂,收受服务对象,收受礼金礼品,敏感人士,斯巴大,新疆人大常委会原副主任,新疆生产建设兵团纪委,新疆自治区原党委副书记,新语丝,方lizhi,方绍东,施红辉,旅游费用在南漳县政协机关报销,晏德文,曹务顺,曹晓明,曹濮生,曾庆洪,曾游海,曾黄麟,金道铭,有关系紧密的房地产商或工程老板,未批先建且违反了办公用房装修相关规定,本人或伙同其亲属收受巨额财物,朱中华,朱太中,朱家栋,朱德灿,朱志新,朱榕基,权俊良,李东光,李云彪,李云忠,李云忠 ,李华中,李华林,李双庆,李喜辰,李圣君,李大农,李富山,李庄事件,李建勇,李德文,李拉成,李正昌,李珠江,李立平,李纯德,李荣飚,李葆,李蒲蒲,李长轩,李静丽,杜建华,杜月明,杜浒,杨东升,杨先静,杨姓商人,杨崇友,杨文礼,杨殿钟,杨海震,杨红斌,杨自立,杭州市纪委,林建忠,林志铁,林胜先,柏贵之,柯志敏,柳遂记,梁国英,梁树林,梁棠,梁英林,梁荣好,梅州市中院,梅祖恩,检察机关决定逮捕,樊建峰,欧阳志鸿,正腐,武嵘嵘,武汉市纪委,死八大,死巴大,段建国,殷瑞山,毋保良,母祥玉,毒菜政府,毕广才,毛小平,汕尾市中级人民法院,江Core,江志成,江苏省有关检察机关,江苏省纪委监察

厅,江西省十二届人大常委会,江西省吉安市中级人民法院,江西省委常委,江西省政协副主席,江西省纪委监察厅,汤志明,汤火山,沈阳军区联勤部原部长,沈阳市中级法院,河北省人民检察院,河北省廊坊市中级人民法院,河北省承德市纪委,河北省纪委机关,河北省纪委监察厅,河北省纪检机关,河南焦作市纪委,河南省人大常委会党组书记,河南省人民检察院郑州铁路运输分院,河南省十二届热大常委会,河南省周口市人民检察院,河南省检察院,河南省洛阳市人民检察院,河南省纪委河南省纪委监察厅,河南省纪委河南省高级法院,河南省纪委监察厅,河南省驻马店市中级法院,洪嘉祥,活摘器官,济南军区原副参谋长,济南市纪委监察局网站,浙江省人民检察院,浙江省政协原党组副书记,浙江省杭州市人民检察院,浙江省纪委,海伍德,海军北海舰队副参谋长,海南省人民检察院指定管辖,海南省副省长,海南省委常委,海南省第一中级人民法院,海南省第二中级人民法院,海南省纪委,涉及“房媳”案,涉嫌严重危机,涉嫌严重违法违纪,涉嫌受贿滥用职权,涉嫌受贿索贿,涉嫌受贿贪污,涉嫌受贿及贪污,涉嫌行贿,涉嫌贪污受贿犯罪,涉嫌贪污受贿罪,涉足当地煤矿电瓷等诸多领域,淮南市大通区检察院,淮南市大通区法院,深圳市纪委,深圳市纪委常委会,温秀田,温震,游xing,湖北省军区原副司令员,湖北省军区原司令员 ,湖北省检察机关,湖北省武汉市人民检察院,湖北省纪委常委,湖南省政协副主席,溧水区纪委,滥用职权受贿犯罪,潘宗营,潘晓东,潘银苗,潮州中院,焦云智,焦明启,熊文健,玉林市纪委,王世益,王云埂,王伟华,王佩英,王刚建,王占一,王宏琨,王宗南,王幼元,王建斌,王新中,王术君,王永全,王洪记,王玉军,王玲珑,王碧玉,王福星,王秀春,王连云,王连永,王道富,王银旺,珠海市香洲区法院,瓷器镇,甘肃省人民检察院,甘肃省纪委,田晓东,留党察看二年处分,白佳卉,白培中案,白彦德,白河县委,皮黔生,矮帝,石八大,石家庄市检察机关,石巴大,祁金立,福州市纪委,福建省地震局,私用公车,私自办假案,程春燕,程晓强,童名谦,第二炮兵副政委,索取收受他人贿赂,索取收受巨额贿赂,索取和非法收受他人财物,索取或收受贿赂,索取案件当事人贿赂,索要1000万元好处费,红色恐怖,经淮南市中院,网评猿,罗世红,罗光勤,罗其方,罗勤宏,罗原仁,罗志君,罗红华,翦保平,肖瑞田,肖禧硕,胡利典,胡南乾,胡尔巴乔夫,胡广兵,胥革,腥猪国,腾鑫曜,自治区高级人民法院纪检组,艾末末,苏建勋,苏玉峰,苏茂森,苗永清,范昕建,范有毅,茉莉花革命,葛七宝,葛年生,董洪运,蒋光头,蒙增杰,蒲忠,蔡成礼,蔺晓军,蕲春县监察局,薛永森,薛玉川,蚌埠市中级人民法院,行政警告处分,行政记过处分,袁健淋,袁志刚,袁惠兴,袁贵人,裴仕伟,襄阳市纪委,西安市纪委,西

藏军区副政委,西藏自治区纪委,覃香喜,解先文,解放军总后勤部原副部长,许左,许志永,许润龙,许鹤岷,诬陷上访户强奸罪,谌晓亮,谢克敏,谢兰茂,谢可明,谢祖文,谭建忠,谭建明,谷丽萍,贵州省委常委,贵州省安顺市纪委,贵阳市第十三届人大常委会,贺维林,贾小晔,赖金水,走私卷烟,赵中社,赵兴华,赵同玉,赵尔巴乔夫,赵涌涛,赵焕光,赵野松,超标准接待并饮酒致人死亡,越反越腐,辽宁省政协副主席,进行组织调查,领导干部严禁进入私人会所消费,违反中央八项规定精神失职,违反办案纪律,违反廉洁自律规定,羁押场所,违法发放贷款罪,违规公款赴美国旅游,违规批准报销韩永安出国费用,违规接受宴请收受礼品,违规组织公款旅游,遵义市市委书记,邓家贵,邓总湿,邓矮子,邓艳珠,邢太安,邵毅,邵算良,邹世凌,邹文英,郑乔生,郑明珠,郑治发,郑立奎,郑立波,郑良驹,郭伯雄,郭景池,郭有明,郭正刚,郭焕波,郭玉发,鄂州纪工委,酗酒滋事,重庆市人民检察院,金德志,金道铭,钦州市纪委,银川市政府,长平兴,长期婚外情,闻清良,阳宝华,陆明兴,陆钦华,陈G诚,陈一谘,陈世旺,陈争鸣,陈卓尔,陈增新,陈存善,陈川平,陈智欢,陈杨广,陈汉军,陈章照,陈良纲,陕西政协副主席,陕西省人民检察院指,陕西省朔州市纪委,陕西省纪委,陶金健,随州市监察局,雨伞革命,雨遮革命,零食馆,雾M,霍绍峰,青岛市纪委,青海省纪委,非法收受他人财物,鞠建平,韦丽坤,韦浩国,韩双成,韩学键,韩正元,韩永安,韩跃平,顾逊泉,顿邓,香港中旅,香港觉醒,马伟灵,马文荣,马细东,骆敬泽,高从智,高保平,高新区党工委,高登银,鲁力军,鲁向东,鲁学军,黄冈市监察局,黄华启,黄意,黄梅县监察局,黄洲洲,黄石市纪委,黄羽天,黄道国,黄顺福,龙拨斌,龚红春,赵家人

4.2 Loss Functions

How do neural networks learn how to map predictors to predictions when the function mapping predictors to predictions is unknown? The researcher specifies a *loss function* that represents the distance of the prediction from the 'true' category. A commonly known loss function is squared loss used in OLS regression. Just as in OLS, the goal is to minimize the loss function. For models where the goal is classification, a possible loss function is zero-one

loss where the goal is to minimize the number of incorrect classifications:

$$\ell_{0,1} = \sum_{i=0}^{D} I(\hat{y}_i \neq y_i) \tag{10}$$

Where D is the dataset and I is an indicator function that takes the value of 1 if $f(\hat{y}_i) = y_i$ and 0 otherwise. However, because this loss function is not differentiable,[31] cross-entropy loss is commonly used instead:

$$\textit{Cross-entropy}_{y,\hat{y}} = -\sum_{i=0}^{D} y_i \log \hat{y}_i \tag{11}$$

Using cross-entropy loss makes the goal of the model to minimize the distance between the correct distribution which puts all the probability mass on the correct category (y_i) and the prediction of a model returns a probability that an observation is in a particular category. Essentially, the goal is to get the model to assign a probability of 1 to the correct category for each observation. In practice, this is very similar to minimizing the number of incorrect classifications.

4.3 Word Embeddings

This section expands the explanation of embedding layers from the main text by summarizing information on embedding layer training from TensorFlow (2018). The model shown in equation 1 could be trained by maximizing its log-likelihood on the training set:

$$J_{\mathrm{ML}} = \log P(w_t|h) = \mathrm{score}(w_t, h) - log\left(\sum_{\mathrm{word}}^{\text{vocabulary size}} \exp\{\mathrm{score}(w', h)\}\right) \tag{12}$$

In practice this method is not used because it is computationally expensive and because embedding layers do not require a full probabilistic model. Instead, embedding layers use

[31] The loss function needs to be differenced to calculate the gradient for stochastic gradient descent. The model uses the gradient, the derivative of the loss function with respect to the parameters, to compare results using different parameters.

logistic regression to distinguish real target words w_t from k noise words $\tilde{w}$. For each example, the following equation is maximized:

$$J_{\text{NEG}} = \log Q_\theta(D = 1|w_t, h) + k \mathop{\mathbb{E}}_{\tilde{w} \sim P_{\text{noise}}} [\log Q_\theta(D = 0|\tilde{w}, h)] \tag{13}$$

Here $Q_\theta(D = 1|\tilde{w}, h)$ is the predicted probability from the logistic regression that word w appears in context h in the dataset D, calculated using the learned embedding vectors θ. In practice, the expectation is approximated with a Monte Carlo average, drawing k noise words from the noise distribution. When equation 13 is maximized, the model will assign high probabilities to real words and low probabilities to noise words.

4.4 The Full Model Graphs

The following is the Keras model summary for our LSTM model that classifies the Weibo posts. The layers are in order with the final prediction at the bottom. The embedding layer has 100 dimensions. The dropout layer randomly drops out half of the nodes in the previous layer during training to prevent over-fitting. The bidirectional layer is a bidirectional LSTM layer. It is composed of 2 LSTM models each with 27 hidden units. One reads the document forward, and the other reads the document backward. The prediction of both units is combined with concatenation. Each LSTM unit also contains a dropout layer. The final prediction layer is a fully connected layer (or dense layer in Keras' terminology). This layer uses a sigmoid activation function, making it a logistic regression where the classification is 1 when the post is political and 0 otherwise.

```
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 174, 100)          4002900
_________________________________________________________________
dropout_1 (Dropout)          (None, 174, 100)          0
_________________________________________________________________
bidirectional_1              (None, 54)                27648
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 55
=================================================================
Total params: 4,030,603
Trainable params: 4,030,603
Non-trainable params: 0
_________________________________________________________________
```

The following is the Keras model summary for the LSTM model that classifies the US newspaper articles (truncation after 100 words).

```
Layer (type)                 Output Shape              Param #
=================================================================
embedding_1 (Embedding)      (None, 101, 100)          7234400
_________________________________________________________________
dropout_1 (Dropout)          (None, 101, 100)          0
_________________________________________________________________
bidirectional_1              (None, 20)                8880
_________________________________________________________________
dense_1 (Dense)              (None, 1)                 21
=================================================================
Total params: 7,243,301
```

```
Trainable params: 7,243,301

Non-trainable params: 0

_________________________________________________________________
```

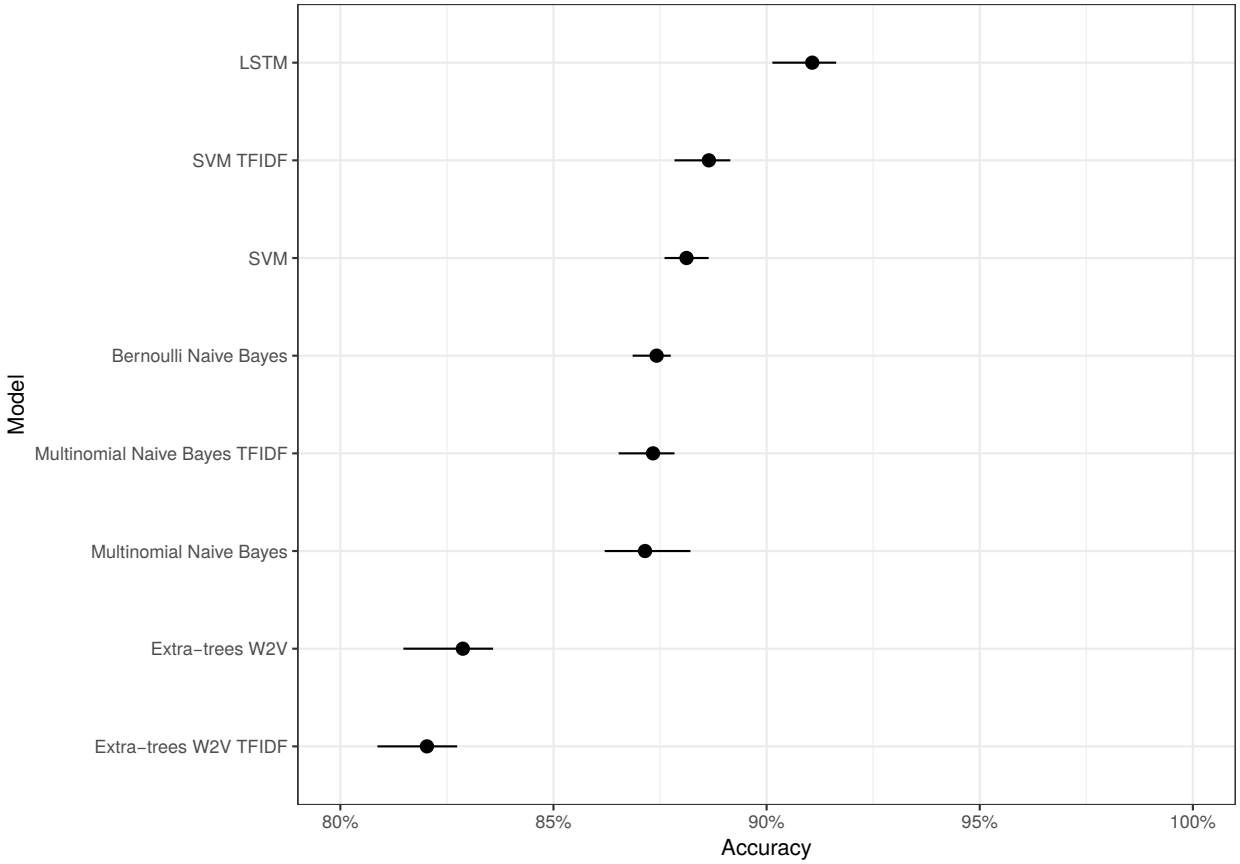
4.5 Figure 1 with No Validation Set

The following figure is the equivalent of Figure 1 except that the LSTM model is set to stop after 11 epochs rather than use a validation set for early stopping.

Figure 3 shows the LSTM model continues to outperform them with a mean score approximately 2.5 percentage points higher than SVM TFIDF's mean score and a minimum score higher than SVM TFIDF's maximum score.

Table 4.5 is the same as Table 3.5 except the LSTM scores are reported from the model without a validation set described above. The LSTM still achieves superior precision and recall.

Figure 3: Comparing Accuracy



The bars show the minimum and maximum scores for 5 draws of train and test sets from the data while the points show the mean score. The full dataset is 10,691 posts. The models use a balanced training dataset of 8,552 posts or 80% of the full dataset. All models use 2,139 posts in the test set. The hyper-parameters for each model are tuned independently.

Table 6: Precision and Recall

Model	Precision	Recall
LSTM	0.931	0.891
	(0.012)	(0.018)
SVM TFIDF	0.887	0.887
	(0.006)	(0.006)
SVM	0.882	0.881
	(0.005)	(0.005)
Bernoulli Naive Bayes	0.874	0.874
	(0.003)	(0.003)
Multinomial Naive Bayes TFIDF	0.874	0.873
	(0.005)	(0.006)
Multinomial Naive Bayes	0.873	0.871
	(0.007)	(0.007)
Extra-trees W2V	0.829	0.829
	(0.008)	(0.008)
Extra-trees W2V TFIDF	0.821	0.820
	(0.007)	(0.007)

The mean score from 5 train and test set draws are show with standard errors in parentheses.

4.6 LSTM Model Implementation Advice

If you use our code templates available at `https://doi.org/10.7910/DVN/MRVKIR`, many of the decisions outlined here are already implemented. However, you will want to read section 4.6.1 because you will still want to tune your model to your particular task as well as section 4.6.4 that discusses GPU vs CPU implementation. Readers also may wish to consult the other subsections below if they wish to understand further why we made some of the decisions we did, deviate from the templates, or understand more about how LSTM models work.

4.6.1 Tuning

As machine learning models, such as SVM, that have tunable parameters have been used in political science before, the purpose of this section is not to explain how to go about tuning machine learning model parameters in general. However, we do provide code templates to facilitate hyperparameters tuning at `https://doi.org/10.7910/DVN/MRVKIR`. Instead, this

section is designed to point out information about LSTM models that is useful to know when making decisions about how to search for optimal parameters.

The first difference between tuning LSTM models and traditional models like SVM is that LSTM models have many more tunable parameters.[32] While this might seem overwhelming, research has shown that the two most important tunable parameters are the learning rate and the number of hidden units (Greff et al. 2016). Further, while these two parameters have the largest interaction with each other of any of the parameters, this interaction is small, which "implies that the hyperparameters can be tuned independently" (Greff et al. 2016).

We recommend the use of validation and early stopping because they free the researcher from needing to select a number of epochs in advance. Dropout layers are used to reduce overfitting, and while they can be set to any probability, a probability of 0.5 is recommended as "close to optimal for a wide range of networks and tasks" (Srivastava et al. 2014, p. 1930). Dropout typically works better than other regularization methods (Srivastava et al. 2014), and we recommend not resorting to other forms of regularization unless a dropout layer has been added after both the embedding and neural network layers without reducing overfitting to an acceptable level. Batch size generally makes little to no difference in performance (Breuel 2015).

4.6.2 Selecting the Optimizer

The optimizer determines how the loss function is minimized. Section 2.2 explains this process in terms of stochastic gradient descent (SDG) because SDG was the original way to achieve this and because modern optimizers are all extensions or variations of SDG. In practice traditional SDG is rarely used because newer variants converge faster or are more tailored to a particular task.

The Keras package documentation recommends the RMSprop optimizer for recurrent

[32]The primary tunable parameters being: the learning rate, number of hidden units, batch size, number of epochs, dropout, kernel regularization, and activation function regularization.

neural networks like LSTM.[33] This is the optimizer we use in the paper and the code templates. One downside of RMSprop is that it is nondeterministic, meaning that even setting a random seed will not guarantee training will produce exactly the same results each time. However, as discussed in section 4.6.3, this is only one of several issues that prevent exact training replicability using these models, and performance should generally be similar enough that others can evaluate whether or not your model is attaining an accuracy score that is within the range you report.

4.6.3 Replicability of Scores

Before getting into a discussion of replicability, it is important to note that all of the issues discussed here refer to sources of variation within the *training* of a neural network. The predictions of a trained network should always be the same. If you use the code we provide at `https://doi.org/10.7910/DVN/MRVKIR` to save a trained version of your model, then you can share this saved model with other researchers who can run the model and get the same predictions. This along with reporting multiple cross-validation scores with some measure of uncertainty such as standard error or the range of scores, largely alleviates the issue that while training results are generally similar with repetition, they are usually not exactly the same.

There are several barriers that prevent exact replication of the training process (and, hence, scores in cross-validation). However, accuracy should generally be in the same range when switching from computers, meaning it should be close enough to verify the plausibility of the reported scores. It is partly for this reason that accuracy scores should always be reported from a series of cross-validation scores rather than a single shot evaluation on a test set. Setting a random seed can help but will not prevent this issue. Some examples of differences between machines that can affect scores include: Whether a GPU or CPU is used for training and which GPU libraries are installed on a computer.

[33] See `https://keras.io/optimizers/` for more information.

Further, nondeterministic processes are usually involved in the training of the networks themselves, preventing results from being precisely the same even when repeatedly run on the same machine. For further discussion of this in the context of TensorFlow with comments from some of the contributors of the package, see `https://github.com/tensorflow/tensorflow/issues/16889`.

4.6.4 GPU vs. CPU

If you have a compatible NVIDIA GPU available for training, we recommend using it rather than a CPU because it will be faster. The good news for users of Keras and TensorFlow is that you do not need to change your code or our code templates at all to switch from one to the other. Whether you have the GPU or CPU version of TensorFlow installed and set at the backend for Keras is what determines whether your computer uses the CPU or GPU to train.

Check `https://developer.nvidia.com/cuda-gpus` to see if your GPU has the required compute power of 3.5 or greater. See `https://www.tensorflow.org/install/gpu` for instructions on installing the GPU version of TensorFlow as well as its prerequisites.